✉️

03. 신경망 학습

신경망 학습 : 경험 데이터로부터 데이터에 내재한 정보와 규칙을 찾아 추론 능력을 만드는 과정

- 반복적으로 최적해를 찾는 최적화 방식으로 이루어짐
  - 경사 하강법
  - 역전파 알고리즘
  - 데이터셋 구성 방식
  - 데이터의 입력 단위
  - 오차 최소화 관점 손실 함수 유도
  - 최대우도추정 관점 손실 함수 유도

3.1 신경망 학습의 의미

▼ 정리

신경망은 입력 데이터가 들어와도 어떤 출력을 만들어야 할지 알 지 못함.

그 규칙을 학습 데이터를 이용해 스스로 찾아내야 함.

신경망이 학습한다 = 규칙을 찾는 과정

신경망의 요소들은 함수적 매핑 관계의 부품과 같은 역할을 함

가중 합산 + 활성 함수 → 뉴런

뉴런 + 뉴런 → 계층

계층 + 계층 → 신경망의 계층 구조

신경망의 구조와 관련된 것들은 학습 전에 미리 정해둠. (하이퍼파라미터)

학습 과정에서 모델 파라미터의 값을 찾음.

⇒ 인공 뉴런의 구조는 사전에 결정하고, 학습 과정에서 뉴런의 연결 강도를 포함한 모델 파라미터 조절.

모델 파라미터 (Model Parameters)

: 모델이 학습을 통해 자동으로 학습하는 값

- 가중치
- 편향
- 특징
    - 데이터에 의존하여 학습 중 변경됨
    - 모델 구조에 따라 수가 결정됨
    - 학습이 끝나면 모델의 상태로 저장됨

하이퍼 파라미터 (Hyper Parameters)

: 학습 전에 사람이 직접 설정하는 값들로, 모델이 스스로 학습하지 않음.

- 활성 함수 관련
    - 시그모이드, ReLU 등등 선택
    - 만약 활성 함수 내부에 학습 가능한 파라미터(PReLU)가 있으면 해당 파라미터는 모델 파라미터
- 모델 구조 관련
    - 깊이
    - 너비
    - 필터 수
    - 커널 크기
- 정규화 관련
    - Dropout 확률
    - BatchNorm의 모멘텀 등
        - mean, variance는 모델 파라미터로 저장됨
- 학습 관련

- 학습률
- 옵티마이저 종류
- 배치 크기
- epoch 수

입출력의 매핑 규칙에서 학습해야 할 것들

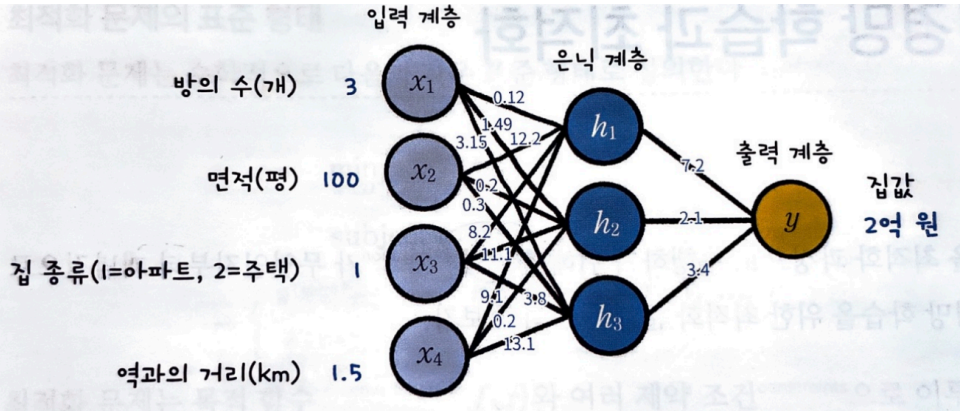
집값 예측 문제가 있다고 가정함.

'방의 수, 면접, 집 종류, 역과의 거리' 데이터가 입력되었을 때 '집값'을 예측한 규칙을 학습을 통해 만들려고 할 때 이 규칙이 만들어지면 모델은 추론 능력이 생겼다고 볼 수 있음.

이러한 규칙은 신경망을 구성하는 모든 뉴런의 가중치와 편향이 결정될 때 완성됨.

→ 학습 과정에서 정확한 값을 예측하도록 신경망 모델의 가중치와 편향 조절

→ 최적의 값이 결정되면 모델은 집값 예측이 가능한 추론 능력을 가지게 됨.



어떤 방법으로 최적의 파라미터 값을 찾아내는 것일까?

→ **최적화 기법 사용**

최적화 기법

: 함수의 해를 근사적으로 찾는 방법

신경망이 관측 데이터를 가장 잘 표현하는 함수가 되도록 만듦.

신경망을 학습한다는 말은 모델의 파라미터값을 결정한다는 의미로 모델 파라미터으 대부분은 뉴런의 가중치와 편향이다.

모델의 파라미터값이 결정되면 신경망에 입력이 들어왔을 때 어떤 출력을 만들어야 할지에 관한 규칙이 함수적 관계로 표현된다.

3.2 신경망 학습과 최적화

▼ 정리

최적화란?

유한한 방정식으로 정확한 해를 구할 수 없을 때 근사적으로 해를 구하는 방법

다양한 제약 조건을 만족하면서 목적 함수를 최대화하거나 최소화하는 해를 반복하여 조금씩 접근하는 방식으로 찾아가는 방법

최적화 문제의 표준 형태

$$\begin{aligned} &\min_{\mathbf{x} \in \mathbb{R}^n} && f(\mathbf{x}) \\ &\text{subject to} && g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, m \\ &&& h_j(\mathbf{x}) = 0, \quad j = 1, \ldots, p \end{aligned}$$

- 목적 함수 f(x)
- 변수 x
- 부등식 형태 제약 조건 g(x) ≤ 0
- 등식 형태 제약 조건 h(x) = 0

표준 최적화 문제 = 변수 x에 대한 등식과 부등식으로 표현되는 여러 제약 조건을 만족하면서 목적 함수인 f(x)를 최소화하는 x를 찾는 문제

최적화를 통해 찾은 x의 값 : 최적해 (optimal solution)

최적해에 점점 가까이 가는 상태 : 수렴한다

최적해를 찾음 : 수렴했다

최소화, 최대화 문제의 관계

문제를 잘 표현할 수 있는 방식으로 골라서 정의하면 됨.

최소화로 표현한 문제는 최대화로 바꾸기 쉽고, 반대도 적용됨.

- 최소화 문제에서의 목적 함수 : 비용 함수(cost function), 손실 함수(loss function)
- 최대화 문제에서의 목적 함수 : 유틸리티 함수(utility function)

신경망 학습을 위한 최적화 문제 정의

- 회귀 문제

  : 타깃과 예측값의 오차를 최소화하는 파라미터 찾기

$$\min_{\theta} \quad \frac{1}{N}\sum_{i=1}^{N}\left(f(\mathbf{x}_i;\boldsymbol{\theta})-t_i\right)^2$$

  세타 - 파라미터, t - 관측 레이블, f() - 모델 예측

  - 손실 함수 : 평균제곱오차(MSE)
    - MSE(mean square error)

      : 타깃과 예측값의 오차를 나타냄

- 분류 문제

  : 관측 확률분포와 예측 확률분포의 차이를 최소화하는 파라미터 찾기

$$\min_{\theta} \quad -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}\log \hat{y}_{ik}$$

  세타 - 파라미터, t - 관측 레이블, log y - 모델 예측

  N - 데이터 포인트 개수, K - 클래스 개수

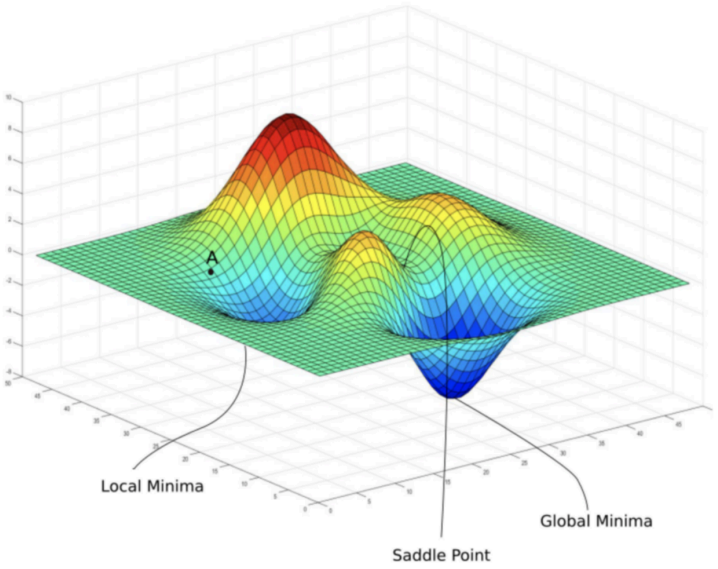
- 손실 함수 : 크로스 엔트로피(cross entropy)
  - 타깃의 확률분포와 모델 예측 확률분포의 차이를 나타냄

최적화를 통한 신경망 학습

최적화 문제 정의 후 최적화를 통해 신경망 학습을 수행함.

최적화 알고리즘은 어느 위치에서 출발하든 손실 함수의 **최소 지점으로 가야 함.**

최적화 알고리즘마다 최적해가 있을거라 예상하는 방향, 이동 폭이 달라짐.



신경망 학습은 최적화를 통해 실행된다.

신경망 학습을 최적화 문제로 정의하면 회귀 문제의 손실 함수는 평균제곱오차(MSE)로 정의되며, 분류 문제의 손실 함수는 크로스 엔트로피로 정의된다.

3.3 경사 하강법

▼ 정리

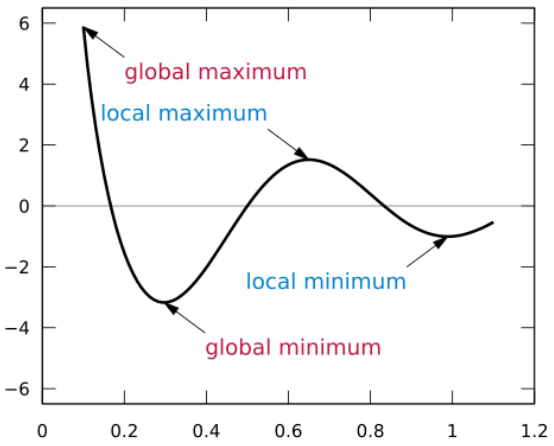
신경망 학습 목표

지역 최소를 찾는 것

전역 최소 (global minimum) : 함수 전체에서 가장 낮은 곳 (최대 / 최소)

지역 최소 (local minimum): 함수에서 부분적으로 낮은 곳 (극대 / 극소)

→ 지역 최소는 손실 함수에 무수히 많음



전역 최소는 곡면의 전체 모양을 확인해야 하므로 계산 비용이 높음

문제가 크고 복잡할 경우 전역 최소를 찾기 어렵고 불가능할 수 있음.

⇒ 지역 최소 찾기

좋은 지역 최소를 찾기 위해 해를 여러 번 찾아 그 중 가장 좋은 해를 선택하거나 동시에 여러 해를 찾아 함께 고려하기도 함

신경망 학습을 위한 최적화 알고리즘

일반적인 최적화 문제는 열린 형태

- 닫힌 형태 : 유한개의 방정식으로 명확한 해를 표현할 수 있는 문제
- 열린 형태 : 유한개의 방정식으로 명확한 해 표현 불가능

열린 형태이기 때문에 미분해서 최대, 최소를 구할 수 없음

→ 조금씩 해에 접근해 가는 방식을 취함

- 최적화 알고리즘

  : 손실 함수 곡면을 근사하는데 사용하는 미분의 차수에 따라 나뉨

  - 1차 미분
    - 경사 하강법
    - 경사 하강법의 변형 알고리즘 : SGD, SGD 모멘텀, AdaGrad, RMSProp, Adam
    - 상대적으로 수렴 속도는 느리지만 손실 함수 곡면이 볼록하지 않아도 최적해를 찾을 수 있음.

      → 손실 함수 곡면이 매우 복잡한 신경망에서 안정적으로 사용하기 좋음
  - 1.5차 미분
    - 준 뉴턴 방법
    - 켤레 경사 하강법
    - 레벤버그-마쿼트 방법
    - 1차 미분을 이용해 2차 미분을 근사하는 방식

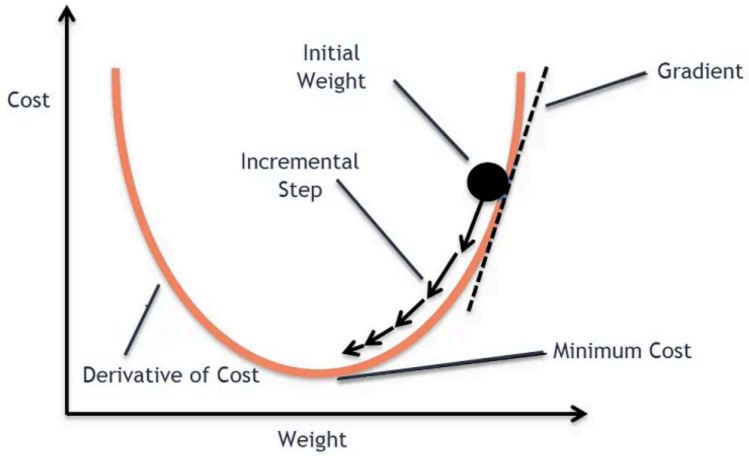
      → 최적해를 빠르게 찾을 수 있음

      → 하지만 2차 미분 근사 알고리즘을 실행해야 하므로 메모리 사용량 높음
  - 2차 미분
    - 뉴턴 방법
    - 내부점법
    - 곡률을 사용해 최적해를 빠르게 찾을 수 있음

      → 손실 함수 곡면이 볼록해야만 찾을 수 있고, 계산 비용과 메모리 사용량 높음

      → 신경망에서 사용하기 어려움

경사 하강법 (gradient descent)



$$\theta \leftarrow \theta - \alpha \cdot \frac{\partial L}{\partial \theta}$$

→ 업데이트 식

θ - 신경망 모델의 파라미터

L - 손실 함수

: 손실 함수의 최소 지점을 찾기 위해 경사가 가장 가파른 곳을 찾아서 한 걸음씩 내려가는 방법

점점 내려가다 보면 결국 최소 지점에 도달할 것이라고 가정하는 방식

파라미터 업데이트 과정을 반복하다가 임계치 이하가 되면 최소 지점에 도달한 것으로 판단, 이동을 멈춤.

α : 스텝 크기 / 학습률 → 이동 폭 결정

$-\frac{\partial L}{\partial \theta}$: 이동 방향. 기울기의 음수 방향을 나타내므로 현재 지점에서 가장 가파른 내리막길로 내려가겠다는 의미

$\frac{\partial L}{\partial \theta}$: 손실 함수 L을 파라미터 θ에 대해 미분한 그레이디언트

gradient : 실수 함수의 미분. x에서 함수 f(x)가 증가하는 방향과 증가율을 나타냄

신경망에 경사 하강법 적용

- 2계층 신경망 회귀 모델
  - 은닉 계층 활성 함수 : ReLU
  - 출력 계층 활성 함수 : 항등 함수
  - 손실 함수 : MSE

이건 책을 읽어야 할듯

신경망의 기본 최적화 알고리즘인 경사 하강법은 손실 함수의 최소 지점을 찾기 위해 경사가 가장 가파른 곳을 찾아서 한 걸음씩 내려가는 방법이다.

신경망은 합성 함수이므로 신경망에 경사 하강법을 적용할 때는 합성 함수의 미분법인 연쇄 법칙을 사용한다.